Prolog:

1. family

female(pam).

female(liz).

female(pat).

female(ann).

male(jim).

male(tom).

male(bob).

male(peter).

parent(pam,bob).

parent(tom,bob).

parent(tom,liz).

parent(bob,ann).

parent(bob,pat).

parent(pat,jim).

parent(bob,peter).

parent(peter,jim).

mother(X,Y):-parent(X,Y),female(X).

father(X,Y):-parent(X,Y),male(X).

grandmother(X):-parent(X,\_).

sister(X,Y):-parent(Z,X),parent(Z,Y),female(X),X\==Y.

brother(X,Y):-parent(Z,X),parent(Z,Y),male(X),X\==Y.



2. diet control

/\*Simple disease\*/

domains

disease,indication,name = symbol

predicates

hypothesis(name,disease)

symptom(name,indication)

clauses

symptom(amit,fever).

symptom(amit,rash).

symptom(amit,headache).

symptom(amit,runn\_nose).

symptom(kaushal,chills).

symptom(kaushal,fever).

symptom(kaushal,hedache).

symptom(dipen,runny\_nose).

symptom(dipen,rash).

symptom(dipen,flu).

hypothesis(Patient,measels):-

symptom(Patient,fever),

symptom(Patient,cough),

symptom(Patient,conjunctivitis),

symptom(Patient,rash).

hypothesis(Patient,german\_measles) :-

symptom(Patient,fever),

symptom(Patient,headache),

symptom(Patient,runny\_nose),

symptom(Patient,rash).

hypothesis(Patient,flu) :-

symptom(Patient,fever),

symptom(Patient,headache),

symptom(Patient,body\_ache),

symptom(Patient,chills).

hypothesis(Patient,common\_cold) :-

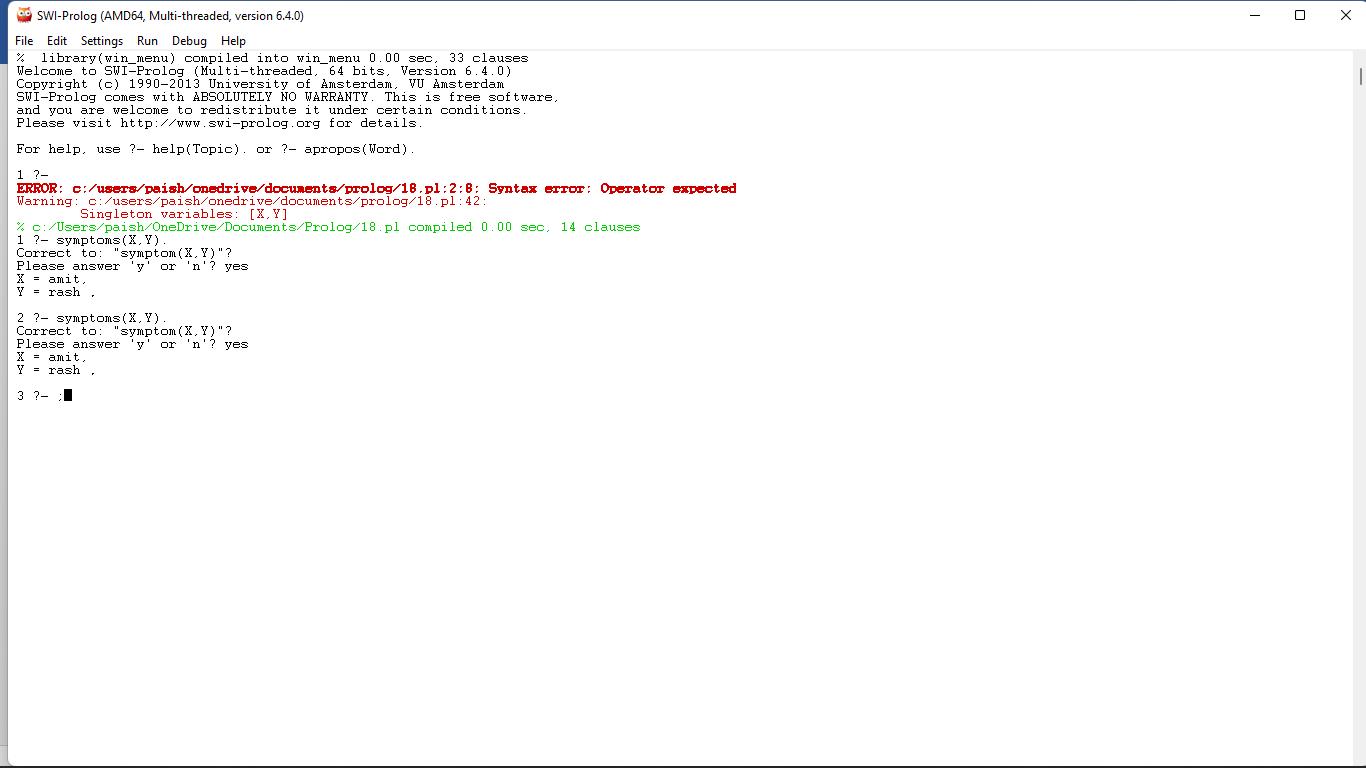
symptom(Patient,headache),

symptom(Patient,sneezing),

symptom(Patient,sore\_throat),

symptom(Patient,chills),

symptom(X,Y).



3.monkey problem:

move(state(middle,onbox,middle,hasnot),

grasp,

state(middle,onbox,middle,has)).

move(state(P,onfloor,P,H),

climb,

state(P,onbox,P,H)).

move(state(P1,onfloor,P1,H),

drag(P1,P2),

state(P2,onfloor,P2,H)).

move(state(P1,onfloor,B,H),

walk(P1,P2),

state(P2,onfloor,B,H)).

canget(state(\_,\_,\_,has)).

canget(State1) :-

move(State1,\_,State2).

canget(State2).

canget(state(atdoor, onfloor, atwindow, hasnot)).



4.colour of fruit:

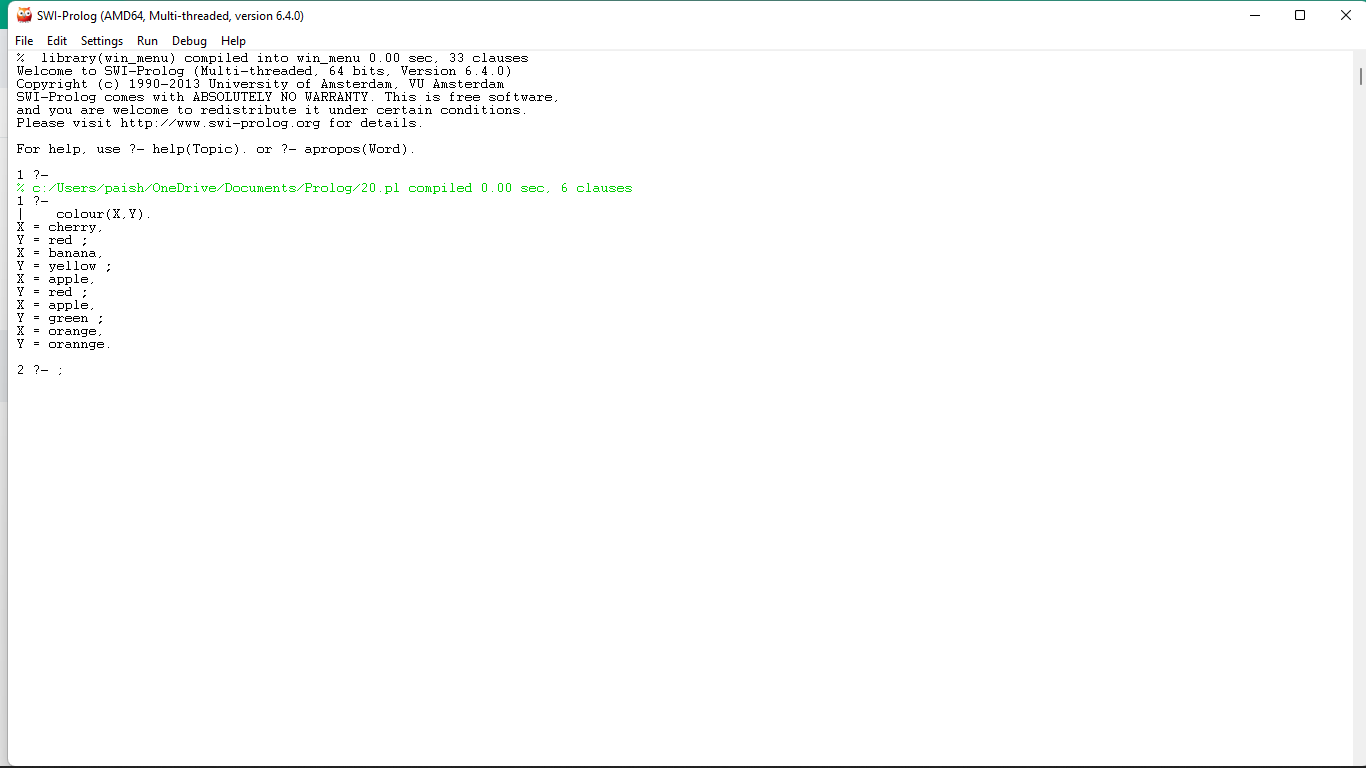
colour(cherry, red).

colour(banana, yellow).

colour(apple, red).

colour(apple, green).

colour(orange, orannge).



5.backword chaining:

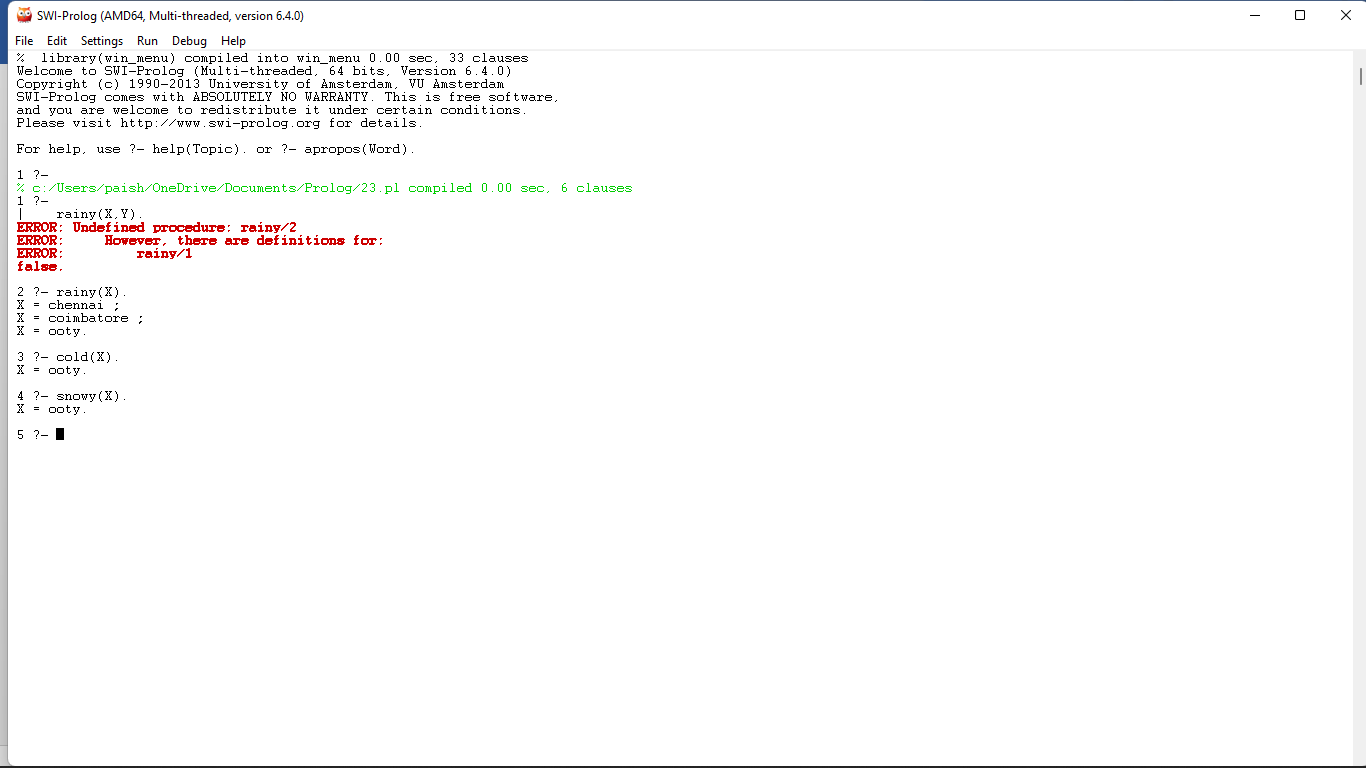
rainy(chennai).

rainy(coimbatore).

rainy(ooty).

cold(ooty).

snowy(X):-rainy(X),cold(X).



6.breadth first search

%connected(+Start, +Goal, -Weight)

connected(1,7,1).

connected(1,8,1).

connected(1,3,1).

connected(7,4,1).

connected(7,20,1).

connected(7,17,1).

connected(8,6,1).

connected(3,9,1).

connected(3,12,1).

connected(9,19,1).

connected(4,42,1).

connected(20,28,1).

connected(17,10,1).

connected2(X,Y,D) :- connected(X,Y,D).

connected2(X,Y,D) :- connected(Y,X,D).

next\_node(Current, Next, Path) :-

connected2(Current, Next, \_),

not(member(Next, Path)).

breadth\_first(Goal, Goal, \_,[Goal]).

breadth\_first(Start, Goal, Visited, Path) :-

findall(X,

(connected2(X,Start,\_),not(member(X,Visited))),

[T|Extend]),

write(Visited), nl,

append(Visited, [T|Extend], Visited2),

append(Path, [T|Extend], [Next|Path2]),

breadth\_first(Next, Goal, Visited2, Path.

7.disease

disease,indication,name = symbol.

predicates

hypothesis(name,disease).

symptom(name,indication).

clauses

symptom(amit,fever).

symptom(amit,rash).

symptom(amit,headache).

symptom(amit,runn\_nose).

symptom(kaushal,chills).

symptom(kaushal,fever).

symptom(kaushal,hedache).

symptom(dipen,runny\_nose).

symptom(dipen,rash).

symptom(dipen,flu).

hypothesis(Patient,measels):-

symptom(Patient,fever),

symptom(Patient,cough),

symptom(Patient,conjunctivitis),

symptom(Patient,rash).

hypothesis(Patient,german\_measles) :-

symptom(Patient,fever),

symptom(Patient,headache),

symptom(Patient,runny\_nose),

symptom(Patient,rash).

hypothesis(Patient,flu) :-

symptom(Patient,fever),

symptom(Patient,headache),

symptom(Patient,body\_ache),

symptom(Patient,chills).

hypothesis(Patient,common\_cold) :-

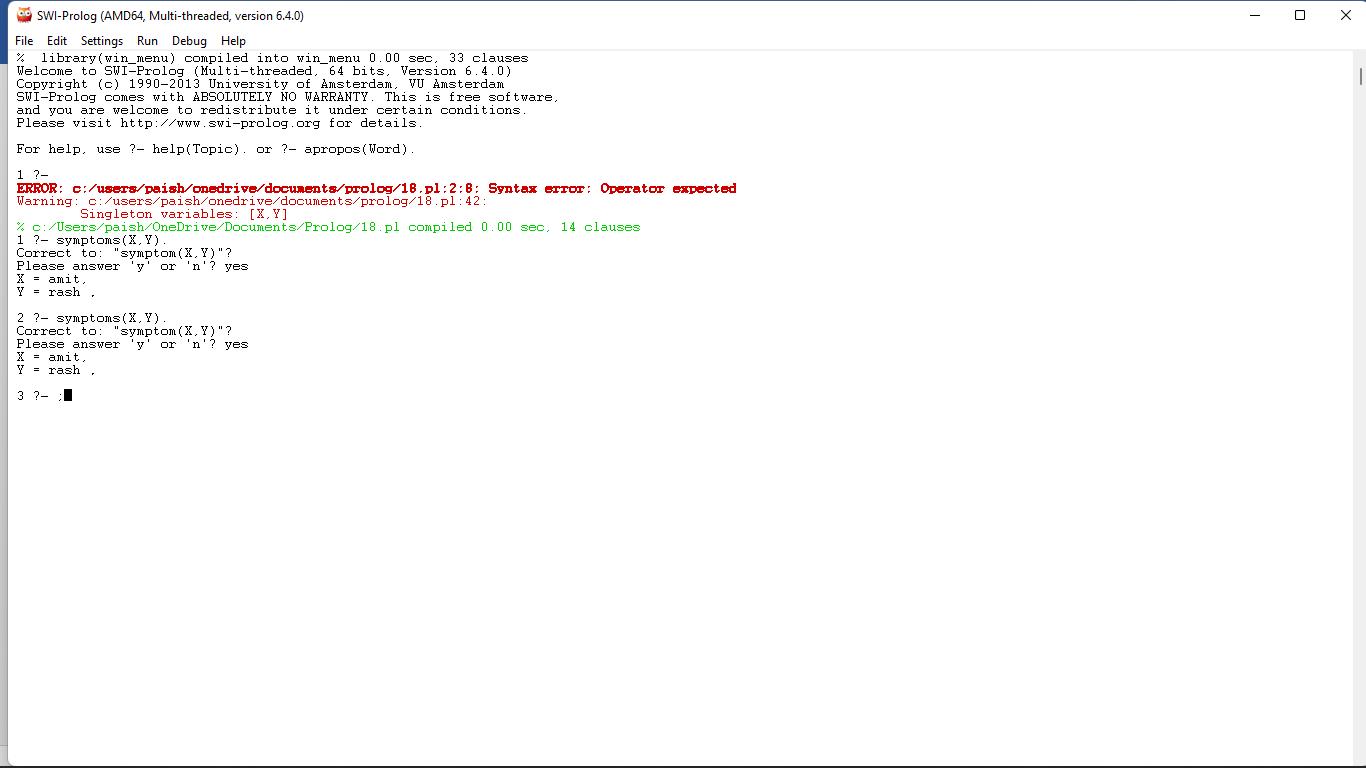
symptom(Patient,headache),

symptom(Patient,sneezing),

symptom(Patient,sore\_throat),

symptom(Patient,chills),

symptom(X,Y)



8. Forward traking:

rainy(chennai).

rainy(coimbatore).

rainy(ooty).

cold(ooty).

snowy(X):-rainy(X),cold(X).

output:

